AFRL-IF-RS-TR-2004-177
Final Technical Report
June 2004

COMMAND AND CONTROL EMBEDDED TRAINING: VISUALIZATION OF THE JOINT BATTLESPACE

Iowa State University

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

AIR FORCE RESEARCH LABORATORY
INFORMATION DIRECTORATE
ROME RESEARCH SITE
ROME, NEW YORK

STINFO FINAL REPORT

This report has been reviewed by the Air Force Research Laboratory, Information Directorate, Public Affairs Office (IFOIPA) and is releasable to the National Technical Information Service (NTIS). At NTIS it will be releasable to the general public, including foreign nations.

AFRL-IF-RS-TR-2004-177 has been reviewed and is approved for publication

APPROVED: /s/

TERRANCE A. STEDMAN
Project Engineer

FOR THE DIRECTOR: /s/

JAMES W. CUSACK, Chief
Information Systems Division
Information Directorate

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 074-0188

1. AGENCY USE ONLY (Leave blank)	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	JUNE 2004	Final Aug 00 – Dec 03

4. TITLE AND SUBTITLE
COMMAND AND CONTROL EMBEDDED TRAINING: VISUALIZATION OF THE JOINT BATTLESPACE

5. FUNDING NUMBERS
C - F30602-00-2-0622
PE - 63227F/63789F
PR - 2743
TA - 00
WU - P3

6. AUTHOR(S)
James Bernard, Carolina Cruz-Neira,
Jim Oliver, and Adrian Sannier

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Iowa State University
Ames Iowa 50011-2274

8. PERFORMING ORGANIZATION REPORT NUMBER
N/A

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)
Air Force Research Laboratory/IFSA
525 Brooks Road
Rome New York 13441-4514

10. SPONSORING / MONITORING AGENCY REPORT NUMBER
AFRL-IF-RS-TR-2004-177

11. SUPPLEMENTARY NOTES

AFRL Project Engineer: Terrance A. Stedman/IFSA/(315) 330-2135/ Terrance.Stedman@rl.af.mil

12a. DISTRIBUTION / AVAILABILITY STATEMENT
APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 Words)*
Working with the Air Force Research Lab's Human Effectiveness Directorate and the Iowa National Guard's 133rd Air Control Squadron, a research team at the Iowa State University's Virtual Reality Applications Center have developed an immersive VR system for distributed mission training called the Virtual Battlespace. The Virtual Battlespace is evolving into a useful exercise planning, pre-briefing, and debriefing tool. The Virtual Battlespace allow participants to analyze airspaces and develop scenarios, and then analyze the outcomes of scenarios, isolate particular engagements, and allow for alternate paths in a tree-like structure. The work has been presented at IITSEC 2002 and IITSEC 2003, and has been described in two papers. The first paper was presented at the NATO SCI Symposium, Critical Design Issues For The Human-Machine Interface, held in Prague, in the Czech Republic in May 2003. The second, Command and Control in Distributed Mission Training - An Immersive Approach", was published in the March 2004 Journal of Battlefield Technology.

14. SUBJECT TERMS
Virtual Reality, Battlespace, Distributed Mission Training, Command and Control Embedded Training

15. NUMBER OF PAGES
41

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	UL

TABLE OF CONTENTS

LIST OF FIGURES

SUMMARY

The Virtual Battlespace was created by researchers at ISU's Virtual Reality Applications Center (VRAC) as a platform for experimentation to determine the positive impact that immersion can have on battlespace management training. Under the initial tasking, our team developed a system to allow multiple JSAF simulations accessed via the network to be viewed immersively. Visually realistic terrain was combined with realistic views of individual entities that were generated in real-time from one or more networked JSAF simulations, or replayed from captured DIS packet streams. Under the original tasking, the system was developed for the front wall of VRAC's four-walled immersive room, the C4, but was subsequently adapted to a variety of immersive devices including desktops, low-cost VR projection systems, as well as the range of immersive devices located at the VRAC.

As part of extended tasking, the VRAC Virtual Battlespace was enhanced to integrate realistic and synthetic representations of a battle into a single system, allowing a user to view the same battle from a complete range of perspectives. Users can view a battle from above, from an isometric view, from the cockpit of an entity, or from any other location in the battlespace. To improve the system's usability, in addition to open navigation throughout the space, a wireless collaborative interface was also created enabling multiple users to control battle perspectives and interact with the visualization, allowing users to acquire situational awareness at the scales and vantage points most applicable to their mission.

The VRAC team also successfully extended the battlespace into a multi-user, collaborative system that allows interaction between participants at a variety of levels of immersion. The current version of the battlespace can be used to combine a traditional 2D desktop user, a user at an immersive desk, a fully immersed user in the C4 or C6, and a group of from 10-200 users in the VRAC stereo auditorium into a common situational environment.

During the course of the research program, VRAC has collaborated with Dr. Rebecca Brooks of AFRL, Mesa and Col. Breitbach and the Iowa National Guard's 133rd Air Control Squadron (133rd ACS) to develop effective battle visualizations that capitalize on the advantages of immersion. While this work has been taken in several different directions, input from the 133rd, as well as feedback from Dr. Brooks and visualization experts at AFRL, Rome, identified the investigation of deployable technology as a priority.

Based on these discussions, ISU applied the Virtual Battlespace to specific areas of the Command and Control process, including exercise preplanning, exercise observation and debriefing, and the control of simulated aircraft. To maximize the transferability of this technology, we also successfully developed a version of the battlespace capable of the aforementioned functions using a deployable display system based on commodity projectors and PC based image generators.

Exercise development, pre-briefing and debriefing

In the summer of 1906, Maj. Eben Swift, then the assistant commandant of the General Service and Staff School, traveled by train to Georgia with twelve officer-students at Fort Leavenworth. So began the first "staff ride" for instructors and students at what is now the U.S. Army Command and General Staff College [1]. Staff rides and related activities, such as tactical exercises without troops, are time-honored military training aids that have been in use for many years. Students and instructors stand on high ground viewing town and country, deploying imaginary troops, and envisioning enemy responses. Exercises are set by instructors and then

students present their solutions for comment and discussion by staff and other students. These techniques teach the vital connection between battlefield conditions and tactics.

Modern engagements are no less dependent on a thorough knowledge of the field. However, unlike battles of the Civil War era, where the majority of a battlefield could be envisioned from the highest hill, the modern air battle is fought over thousands of square miles, a landscape described not only by natural features, such as mountains and rivers, but by “invisible” features such as friendly and enemy sensor and radar fields. The Virtual Battlespace can be extremely valuable in this context, allowing war fighters to traverse and analyze a battlefield to develop strategies and tactics prior to an exercise or engagement. Col Curtis Papke, Division Chief of AFRL’s Warfighter Training Research Division suggested this as a powerful new application of this technology, useful not only for visualizing the field, but for capturing knowledge gained there to feed into the planning process as part of the air tasking order creation process. In this spirit, we propose to expand the Virtual Battlespace interface to allow it to be used as part of the scenario creation process.

The virtual environment can help battle managers and the Joint Force Commanders understand how the decisions they made in simulation exercises affected their overall plan. The positioning of forces and the descriptions of their movements over time are not ideally adapted to 2D PowerPoint presentations. Alternatives are needed. We have developed a basic capability that promises to be useful in this context.

Our experience with the Virtual Battlespace leads us to conclude that visual displays can provide situational awareness at the operational level. By providing an accurate and multi-faceted depiction of what warfighters are encountering, commanders and battle managers can make better decisions. An immersive system can be used to augment the individual displays of battle managers and weapons directors, and provide a common focus for situational awareness of the battlefield.

INTRODUCTION

The exhaustive review of prior campaigns, engagements and plans is a staple of military command training. Consider the staff rides common at the turn of the last century [1]. After extensive study of the battle's history and context, instructors and students would physically ride out to a battlefield site to examine the terrain of the field first hand, taking the vantage points of friend and foe, to see for themselves the interplay between ground, objectives and available force that constrain military strategy.

Modern engagements are no less dependent on a thorough knowledge of the field. However, unlike battles of the Civil War era, where the majority of a battlefield could be envisioned from the highest hill in the county, today's battles are fought over thousands of square miles. The battle landscape is now defined not only by natural features, such as mountains and rivers, but by "invisible" features such as friendly and enemy sensors, the threat zones of long-range weapons, and the forest of targets that must be struck precisely to minimize loss of life. Creating consistent and complete mental pictures of this complex environment is one of the tasks of training, whether as part of pre- and post-mission briefing, or as an integral part of command and control of distributed mission training exercises.

The complexity of modern warfare increases as the number of battle assets grows. With this escalating complexity, commanders are handed the increasingly difficult task of maintaining a clear mental picture of the engagement. This fact heralds the need for an improved method of command and control. Many of the same issues faced by modern training tools run parallel to those now faced by command and control in the field. These issues include the visualization of the visible and invisible features of the battle landscape, as well as the coordination of manned and unmanned resources. An example of this complexity can be found in the use of unmanned aerial vehicles (UAV). While the introduction of the UAV provided the armed forces with a powerful new tool, it quickly became apparent that it required a more effective human interface. The desire for one person to manage a swarm of semi-autonomous UAVs demands a new UAV control paradigm. A primary challenge with current UAV control stations is that it is difficult for one person to maintain situational awareness of both the UAVs and manned craft—once again, a visualization problem related to battle resources and their interactions.

We believe that immersive virtual reality (VR) technologies based on recent work at Iowa State University's Virtual Reality Application's Center (VRAC) can be extremely valuable in all three command and control contexts. Such technologies can allow battle managers and war fighters to traverse and analyze the complex information landscape that is the modern battlefield as it unfolds; they can allow trainers to develop strategies and tactics prior to an exercise or training engagement; and they can provide the basis of the control station for UAV swarm management.

Immersive battlespace visualization can fuse information about tracks, targets, sensors and threats into a comprehensive picture that can be interpreted more readily than other forms of data presentation. It is this quality that makes immersive battlespace visualization ideal for these command and control contexts.

Our experience with the Virtual Battlespace suggests that this technology can be useful in exercise planning, pre-briefing, debriefing, and as a live engagement management tool. The remainder of this report describes the design and implementation of Virtual Battlespace, some of its applications to-date, and the future directions that could be taken.

THEATER AIR CONTROL

The Virtual Battlespace was originally conceived to depict a joint battlespace similar to one monitored by a ground based battle manager or weapons director. Our ideas of the battle manager's function and access to information were formed by first hand observation of battle managers and weapons directors of the Iowa National Guard's 133rd Air Control Squadron. The weapons directors of the 133rd use a monitoring station, the AN/TYQ-23 Modular Control Equipment (MCE) which provides the Air Force with a transportable Theater Air Control System, an automated air command and control system for controlling and coordinating the employment of aircraft and air defense weapons. [2].

MCE's can be deployed into a theatre of operations and used to monitor a full range of tactical data links. Four system operator workstations are housed within an MCE's Operation's Module (OM), providing the operator with interface access, a viewable radarscope and telecommunications capability. Sensors and power are external to the OM. The fundamental system element of the MCE is the Operations Module (OM. Figure 1 shows an operator inside the OM, and Figure 2 shows the MCE with two OMs. [2].

Figure 1: Inside the Operations Module.

Figure 2: Modular Control Equipment.

The OMs can be interconnected to provide a deployable command and control capability. The OM's are typically connected to local radars located within a two kilometer radius by fiber optic cable.

Operators use the displays in the OM to identify and monitor friendly, hostile and unknown tracks in real time. Tracks and targets are identified and classified according to type. Operators use the information gained from the displays to build a mental picture of the current operating picture and use this model to identify targets and threats, and communicate with aircraft and with those responsible for battle management. The OM can provide control functions in support of a range of tactical missions including: air defense, counterair, interdiction, close air support, reconnaissance, refueling, search and rescue, and missions other than war. [2]. Observation of several OM based training exercises, together with personal interviews with the trainees and trainers formed the basis for the VRAC team's initial development of the Virtual Battlespace as a tool for representing the theater level engagements the 133rd typically trains for.

METHODS, ASSUMPTIONS AND PROCEDURES

The Virtual Battlespace uses virtual reality immersion display technology along with the fusion of multiple data streams to provide a user with a clear representation of the information needed to understand and control a battle. The Virtual Battlespace system connects users to information streams using a display system and a role-based user interface.

The Virtual Battlespace visualization system is flexible, allowing it to support multiple end uses. Battlespace users need to be able to maintain an understanding of the entire battlefield or scenario, and yet be able to acquire specific details about individual units. To support these requirements, the Virtual Battlespace provides a comprehensive view of the overall field and can provide additional detail as a user narrows his visual focus to a portion of the space. The Virtual Battlespace architecture can also accommodate system nodes that generate data streams associated with individual units, such as pilots in a flight simulator. Virtual Battlespace incorporates these users into a common system, allowing them to interact with one another in a distributed way.

There are many different streams of information that provide support for battlefield decision making. Some of these include radar and other sensor feeds, satellite imagery, communication links, and weapons information. Virtual Battlespace is designed to fuse multiple information streams and make them centrally available to command and control personnel. The goal of this comprehensive presentation is to improve a user's ability to make effective and intelligent decisions [3,4].

A general architecture of the system is shown in Figure 3. In Virtual Battlespace, data streams are separated into two main categories: entity-based data and battle-level information. Entity-based information streams deal with the location, attitude, path, weapons, and sensors for a particular weapons system or entity in the battlespace. This information is needed to give the commander an indication of the assets and threats that are present and to paint a global picture of the overall field. Battle-level streams include: satellite imagery, video feeds of sectors and munitions, and communication networks among units. In Virtual Battlespace, these streams are presented graphically to reduce the amount of textual information presented to the commander allowing them to focus more time on critical decisions. Several research groups have explored using virtual "sand tables" [5–7] to display and interact with such data using large screens projected from beneath.

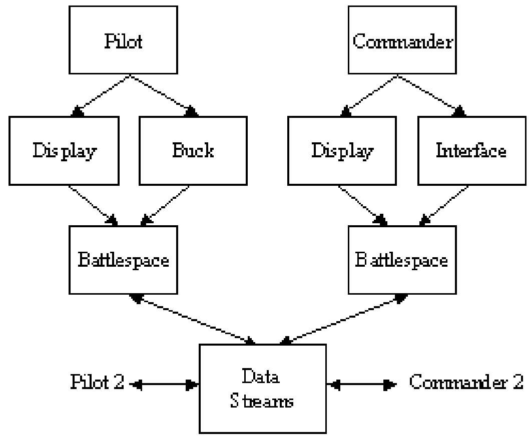

Figure 3. General architecture.

The information streams are made available to the user of Virtual Battlespace through the immersive display system. To make the Virtual Battlespace useful in the widest possible context, the display system is designed to support the complete range of delivery platforms, from permanent, high-end multi-walled immersive projection theatres to lower-cost, deployable systems. With such a design, units with deployable systems in or near the field could be connected with a permanent installation at a central command center to provide a common operating picture.

The user controls information display in Virtual Battlespace with a distributed, cooperative user interface. To avoid information overload and allow the user to tailor information display to meet individual needs, Virtual Battlespace allows users to interact easily with the system and focus solely on the information that they need. By decoupling Virtual Battlespace's user interface from the underlying application, individual users can simultaneously interact with a common application through interfaces specifically tailored to their roles. Using these decoupled interface tools, users can choose the scale and presentation level of information on a common display to highlight particular aspects of the overall engagement. In this way, Virtual Battlespace facilitates not only a user's ability to view and understand the battle but also provides a means to control it.

System Architecture

This section discusses some of the design goals and decisions made in developing Virtual Battlespace. Figure 4 presents a subsystem-level diagram of the system architecture showing the relationships between its major components. In this diagram, the flow of data is from bottom to top. Data streams originate either from a simulator or a mission participant. This scenario data is then sent through the data stream managers to the proxies, and are then displayed to the user.

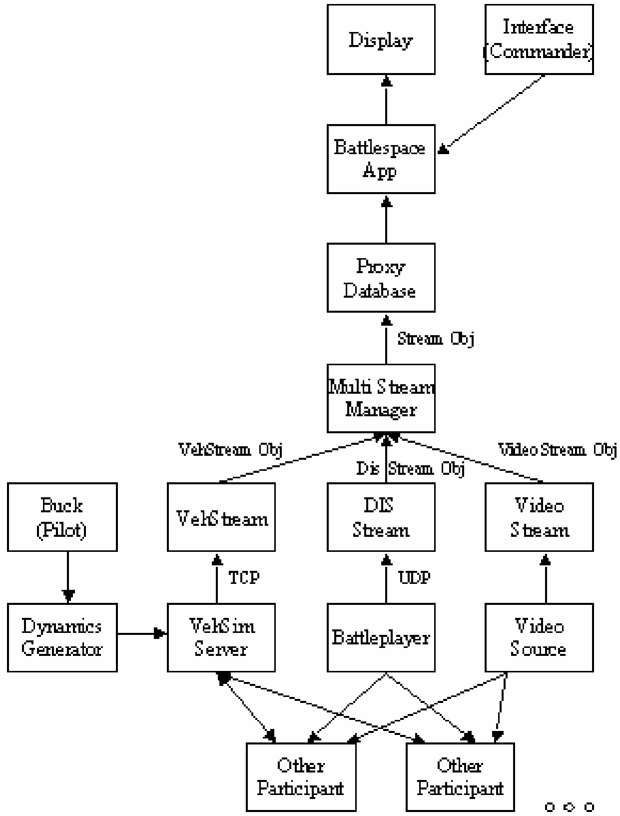

Figure 4: System Architecture

The Battleplayer is responsible for sending out information about the units in the JSAF scenario in DIS format. This creates the DIS stream, an example data stream. An individual data stream is a connection to a data source that produces a time-series of data packets. Another stream is created by the Video Source. This stream is a video feed from a camera at a place of interest for the battle scenario. Another stream is sent out by the VehSim server. The VehSim stream is a locally developed stream of state information for one unit. In a typical case, this unit is an aircraft controlled by a pilot (as shown on the left side of Figure 4) using a flight Dynamics Generator designed here at Iowa State University. Each stream is processed by a specific stream manager to create entities in the proxy database. The stream managers for the three streams are the DIS Stream Manager, the Video Stream Manager and the VehSim Stream Manager. These managers pass the incoming data to the right type of proxy in the Proxy Database. The data proxies encapsulate common interfaces for data types that are displayable within the Virtual Battlespace system. New streams of data are incorporated by specializing one of Virtual Battlespace's defined data proxy interfaces, allowing for stream specific manipulation of entity or battle level data while facilitating its display within the common interaction environment. The proxy interface provides the rest of the system with a common set of object interfaces that insulate the system at large from specific data stream encodings. This approach allows the system to incorporate disparately defined data streams more easily. The proxies are described in more detail in a later section. The Battlespace application uses proxies to draw all of the units in the battle as well as to display video feeds. The operator of the Virtual Battlespace then uses the Tweek Interface to interact with the application. This interface is described in detail later. This architecture, while complex, has proven to be useable and flexible to changes in the system and information that must be incorporated and displayed on the virtual battlefield.

Information Stream Management

Central to Virtual Battlespace is the ability to fuse diverse data streams into an integrated display. This requires a system that allows incorporation of undefined data formats, while simultaneously creating a set of information display tools that can be used to display information from a variety of sources in a common way. Virtual Battlespace could easily be made HLA-compliant through the addition of a component that would subscribe to an HLA-based federation [8]. However, Virtual Battlespace had as a further design goal that the addition of non-HLA streams is easy and straightforward. This goal is achieved through the implementation of an application-level stream manager responsible for integrating multiple data streams and for providing a common set of internal interfaces for data interaction. This critical component, the Multistream manager, manages the process of conversion of raw stream data into stream object data.

Several diverse sources such as a simulated force generator like Joint Semi-Autonomous Forces (JSAF), or a live sensor such as radar feed, or a multimedia signal such as audio or video can generate a stream of data. The streams need not have a common format. The Multistream manager is responsible for fusing these disparate, dynamic streams into a coordinated set of data objects, which can be interfaced in a common way by the rest of the application. In the current implementation of the system, a video stream, a force stream, using the Distributed Interactive Simulation (DIS) communication protocol [8], and a proprietary vehicle simulation stream (VehSim) [9] are fused by the Multistream manager into a coordinated data structure.

The presence of different coordinate systems makes this problem even more difficult. Many streams will have different ways of representing the world. JSAF for example represents the world in terms of XYZ distance from the center of the earth in meters. The VehSim stream however represents the world in terms of distance from an origin relative to the area that the person is flying over. At the start of the program the base coordinate system must be defined and from then on incoming data streams positional and orientation data is converted into the local coordinate system.

The VehSim stream is the output of a human-in-the-loop vehicle simulation containing a time series of vehicle data including position, acceleration, and orientation. The simulator takes the inputs from the human and uses a dynamics engine to generate time-stamped vehicle data. This data is then sent via TCP/IP as the VehSim stream. The VehSim protocol supports a small number of simultaneous vehicles updated at a high frequency. The opposite of this stream in behavior is the force stream. This stream sends DIS packets across a UDP connection and is capable of handling a large number of individual entities, each updated at a low frequency. In Virtual Battlespace, the DIS stream is generated using a JSAF scenario builder and is used to generate the bulk of the battle participants. The final stream implemented is a simple video feed. The Multistream manager allows a video stream to be integrated into the overall time stream, coordinating when and for how long each frame is played, and where it is to appear. The video stream can be either a live video feed, or a series of stored clips.

Even with the work of the Multistream manager there is one more step to achieving the more general architecture in Virtual Battlespace. The data provided by the multi stream manager must be displayed. Taking the information and displaying it from different sources of information are the tasks of the Proxy Database.

Proxy Database

The graphical elements used to display the data streams are a major component of the system. They not only portray the physical attributes of entities in Virtual Battlespace, such as relative position, orientation, status and speed, they also portray derived attributes such as prior and future paths or sensor and threat ranges. To maintain the system's flexibility with respect to the format of the input streams, the display of the data streams are separated from the management of the streams and from the base application.

Entity proxies provide the application with a uniform interface to individual entities, independent of the data stream the proxies were generated and updated from. This means that a proxy generated from a flight simulator stream can be displayed with the same graphical components as an entity generated from a DIS stream. This approach simplifies the interface not only between the application and the proxies, but also between the user and the entities. The user has no direct knowledge of the number of different information streams that are driving the system. All graphical functionality is expressed in terms of a common interface which all entity objects support. This allows entities represented by disparate data streams to be treated uniformly by the remainder of the application.

An example of this approach is in the implementation of the VehSim Proxy and DIS Proxy. The VehSim and DIS streams represent similar information, but at widely differing update rates, referencing different coordinate systems. The proxy implementations for each stream encapsulate the transformation of this information into a common representation and common coordinate system. The base implementation of proxy provides methods to support graphical entity display based on the rest of the proxy interface. However, derived proxies can override these basic definitions to define type specific behaviors if need be.

Another important aspect of the proxy database is that it supports the display of aggregate representations of groups of entities. These aggregate objects suppress the individual entity representations to reduce information overload. This allows, for example, flights of aircraft to be displayed as composite entities to simplify a commander's view of a battle. The recursive nature of the proxy model allows aggregation at arbitrary levels by supporting aggregates of aggregates. Because aggregates derive from Proxy, they can be treated the same by the drawing function. As a result, aggregates can leave trails, have height sticks and slant ranges attached to them, have their sensor and threat ranges displayed, and have a shadow placed below it.

Terrain

A vital component in the Virtual Battlespace system is the terrain. The terrain gives the user the sense of realism and a sense of where he is and what he is doing. Terrain often determines critical battle decisions. Knowledge of the location of a mountain range or river can prove to be a critical piece of information. The early attempts at terrain in this project involved using map information to layout the significant political information needed by the commander. Besides rivers, significant geographical elements were not included. The next step in the terrain generation process was to add geographical elements into the terrain. This was accomplished using United States Geographical Survey data to create an elevation map of the Nellis Air Force base. We chose this data because this is where our scenario was to take place. This data comes in the form of a location on the surface of the earth and its elevation above sea level. From this data we used a terrain generation program provided with Multigen Creator to create the terrain. Then satellite image textures were overlaid on top of these new height map terrain files to create a new terrain. Figure 5 shows this terrain. Relating to the terrain we experimented with atmospheric effects to help the environment feel more real. However upon examining the effect of fog we felt that these effects would only burden the commander's vision of the battlefield. Figure 6 shows the atmospheric fog.

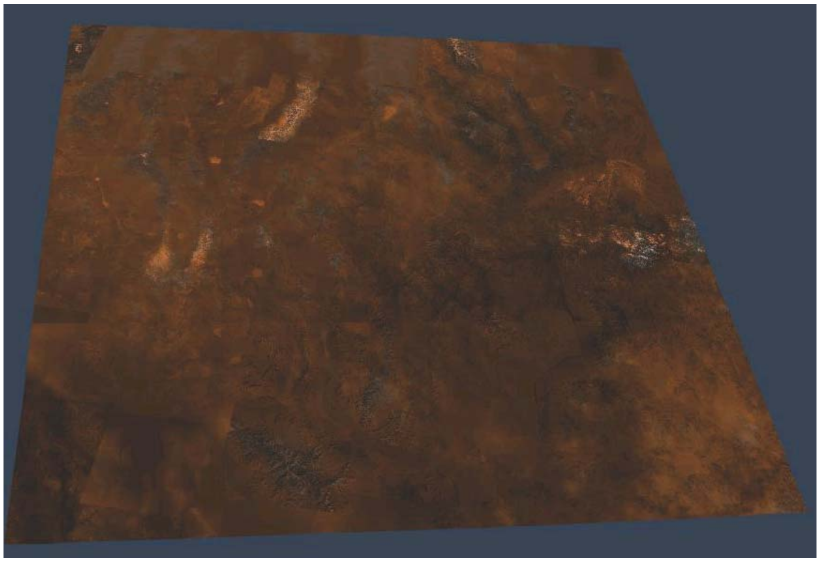

Figure 5: Textured Nellis range

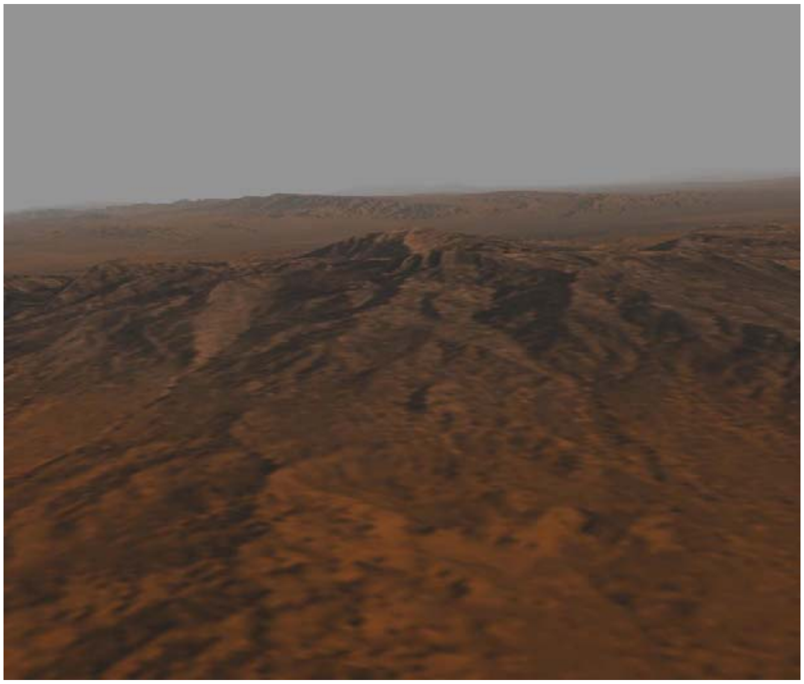

Figure 6: Close-up of terrain detail with fog effect

This method worked reasonably well. However there were some issues with this method. The amount of data used to create the terrain resulted in very high polygon count models that looked relatively flat from high heights. This hurt the performance of the application and didn't emphasize the presence of mountains and geographical effects as much as one would desire.

The final version of the terrain used the satellite images to create the terrain directly. Using a program called Demeter, we took the satellite images we had of the terrain and created dynamic terrain. This dynamic terrain adjusts the level of detail depending on how close the user is to the section of terrain. This means that a high number of polygons are used on parts of the terrain that are close to the user. The program also makes the mountains stand out more. This means that the terrain emphasizes these geographical barriers that can be present in an area. Figure 7 shows this new terrain. Figure 8 shows a fly though view of the terrain. The darkness is simply because the sky has not been added to the world yet.

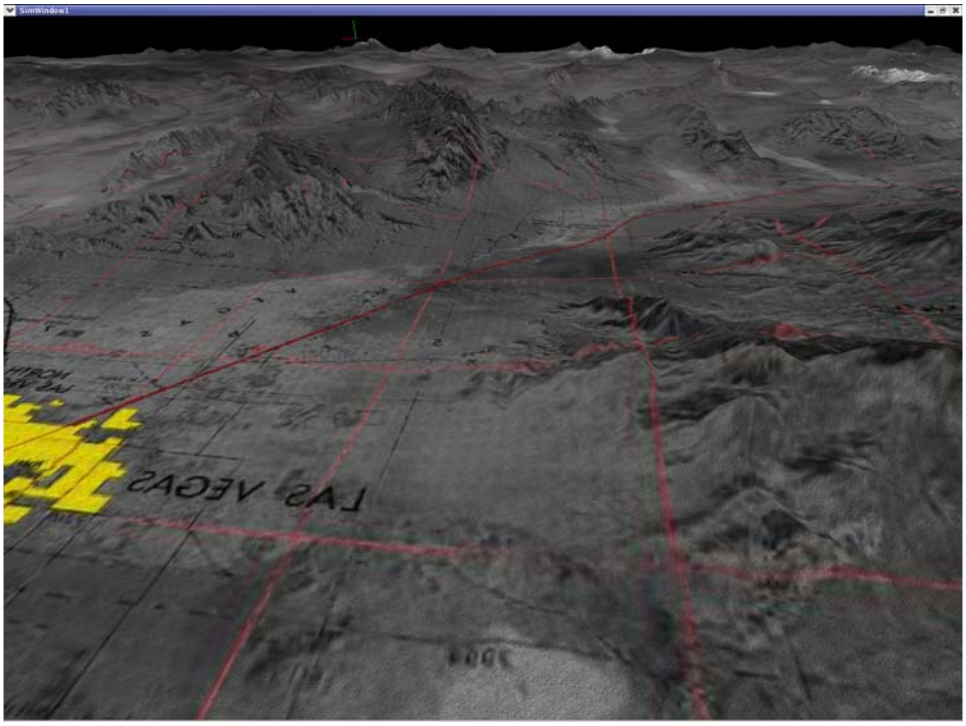

Figure 7: New Nellis Terrain

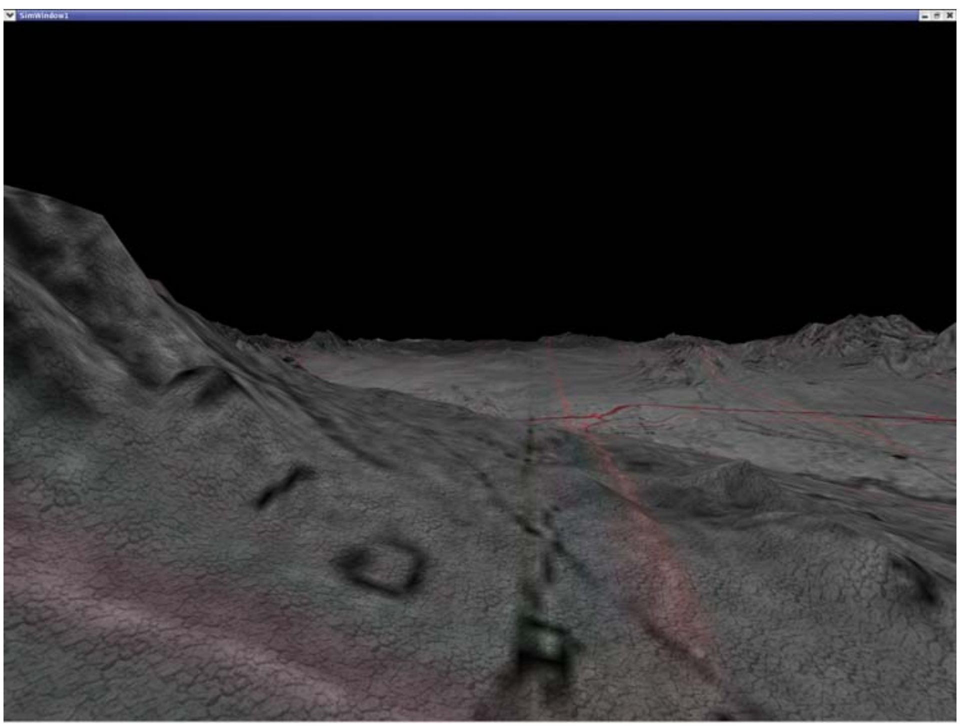

Figure 8: New Nellis Terrain Detail

User Interaction

The typical approach to user interface in immersive applications is a combination of gestural or positional interaction, combined with graphical display cues such as three dimensional menus and selection rays [10]. These interfaces support illusion of immersion by allowing users to interact directly with virtual objects. However, as the complexity of the application increases, the virtual metaphor must be augmented.

For Virtual Battlespace to be effective, users must be able to interact with the simulation to accomplish a wide variety of tasks such as navigation, view scale, aggregation, and selective information display. While some of these tasks are compatible with the usual immersive interface methods, many others are not. The Virtual Battlespace user needs a wide variety of interaction mechanisms that are intuitive yet provide access to a large number of configuration options. Furthermore, while much of the useful information in a battlespace can be conveyed graphically or iconically, sometimes there is simply no substitute for text. In these cases, immersive displays are handicapped because their display resolution is typically not sufficient to display graphics and text simultaneously.

The Virtual Battlespace system uses a combination of two modes of user input. In addition to the gestural navigation and graphical selection interfaces typical of immersive environments, Virtual Battlespace allows participants to interact wirelessly with the simulation via personal interface devices (PDAs, tablet computers, or other Java-capable devices). This is accomplished via an extension to VRJuggler (see below) known as Tweek. Based on CORBA as a remote procedure call mechanism, Tweek allows Java interfaces running on personal interface devices to communicate with the Virtual Battlespace. The Virtual Battlespace registers an interface that allows two-way communication between these devices and the application. Using this interface, Java applications can give remote commands to drive the Virtual Battlespace application or issue queries to obtain status information. Because the interface is decoupled from the application, it is straightforward to provide custom simultaneous interfaces for multiple participants. Figure 9 shows a picture of a Virtual Battlespace's Java interface implemented on a tablet PC via Tweek.

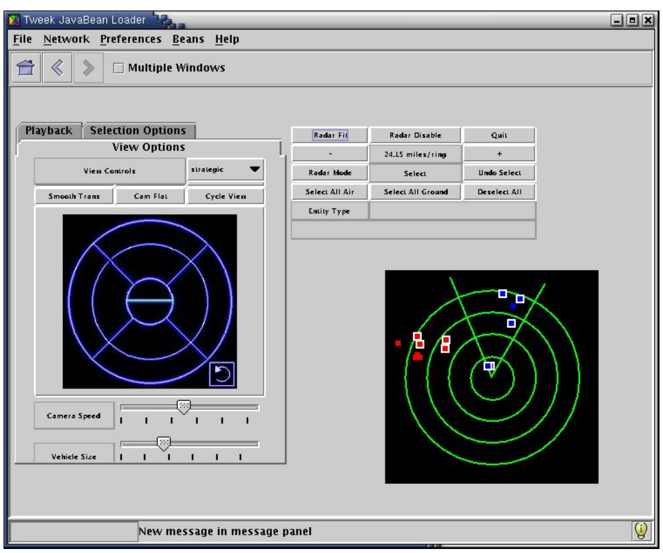

Figure 9: Navigation Controls

With this interface, users can navigate through space, select entities via the interactive radar screen, and perform actions on those entities such as toggling graphical features. Some of these graphical features include height sticks, sensor sweeps, threat zones and heads up displays. Navigation is accomplished by using the large blue circle button on the left side of the interface shown in Figure 9. The x-y plane in the Virtual Battlespace was parallel with the ground plane of the terrain, with the positive x-axis going into the front screen and the positive y-axis to the right. Clicking in the quadrant facing up moved the user in the positive x-direction, while the quadrant facing down moved the user in the negative x-direction. Likewise clicking in the quadrant facing left moved the user in the negative y-direction and the quadrant facing right moved the user in the positive y-direction. Clicking in the outer ring of the circle moved the user faster than clicking on the inner ring. Furthermore, moving the slider directly beneath the navigation circle could increase this movement gain. Finally, by clicking on the small half circle just above the center of the navigation circle, the user would go up and by clicking on the small half circle just below the center of the navigation circle, the user would go down.

This interface not only provides the user the ability to interact with the application but it also provides information to the commander about the Virtual Battlespace. This feedback about the Battlespace is displayed in the radar on the interface. The radar shows where all the units in the engagement are located. The units appear color coded with respect to the color of their team. Furthermore, aggregated units are outlined in white as shown in Figure 9. If no units are outlined in white, as in Figure 10, then it is likely the case that the operator is in pilot mode. In this case, there are many close but separate dots on the radar.

The radar also provides more than just information about the battle. It also functions as a selection device. If the user clicks on a dot, that unit becomes selected in the Virtual Battlespace. With the unit selected, the Toggle Radar button could then be pressed to have the extents of the selected unit’s radar sweep be displayed in the Virtual Battlespace. In the case that the click would select multiple units (especially a concern in pilot mode), the radar screen will automatically zoom up and show all of the units that were in the first select. Then the user must select which unit they wanted and this pattern repeats recursively until only one unit is selected. The radar also shows the user where the view on the front wall of their display device is with respect to the battle. The green wedge shown in Figure 9 gives the extents of the field of view of the front wall. In this way, the user can quickly gain situational awareness about the location of other units. For example, a group of red blips on the Tweek radar to the left of the green wedge will likely show up on the display surface left of the front wall.

Finally, with buttons just above the radar, the user can choose to engage Radar Fit, which will dynamically adjust the extents of the radar to display all units. In this case, the distance in miles between rings on the radar is shown in the text box to the right. If radar fit is toggled off, then the user can increase of decrease this distance manually by using the “+” and “-“ buttons on each side of the distance text box.

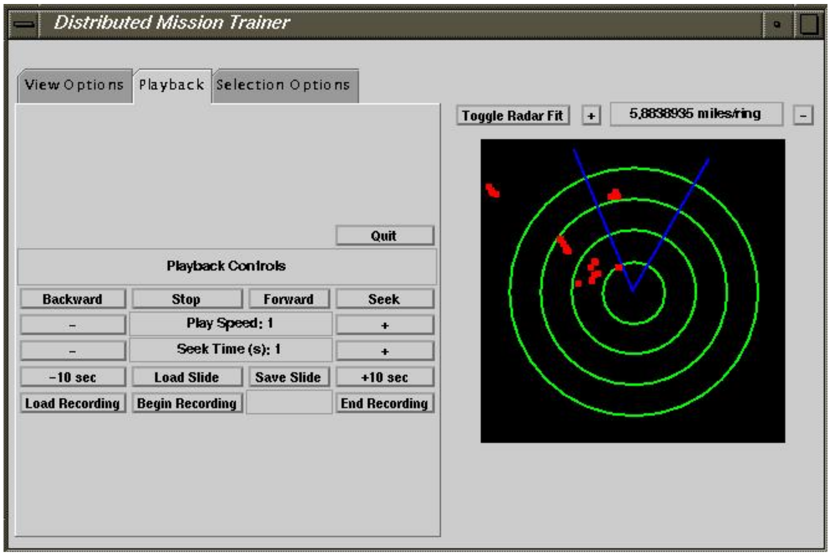

Figure 10: Wireless radar control and coordinated stream VCR controls

Figure 10 shows the record and playback interface for the Virtual Battlespace. This section of the interface is designed to support the generation of content for debriefing meetings. The user can navigate to a point of interest and click “Begin Recording” and red text declaring “REC” will appear in the box next to the button. The recording will save in a generated file name and will save the state of all units in the Virtual Battlespace until “End Recording” is pressed. Previously saved recordings can be loaded with the “Load Recording” button. Once it is loaded, the recording can be played faster or slower by using the “+” and “-“ buttons on either side of the “Play Speed” text box. This box also displays the current play speed. To play the recording, the user presses the “Forward” button and it plays the recording at the play speed. To play the recording backward at the play speed, the user hits the “Backward” button. The user can skip forward or backward through the recording a fixed number of seconds by hitting the “Seek” button. The amount of time skipped is displayed in the “Seek Time” text box. To adjust this time, the user hits the “+” and “-“ buttons next to this text box (or the “-10 sec” and “+10 sec” buttons below them). To seek backward, the user decreases the seek time to a negative number of seconds. Hitting the “Stop” button halts the playback of the recording. These recording playback features are only of interest in a debriefing use of the Virtual Battlespace and are not available in the case that live data is being streamed in. The final options on this screen are “Load Slide” and “Save Slide”. When “Save Slide” is pressed the current state of all units in the Virtual Battlespace at that instant are saved in a file. “Load Slide” retrieves and displays these saved slides.

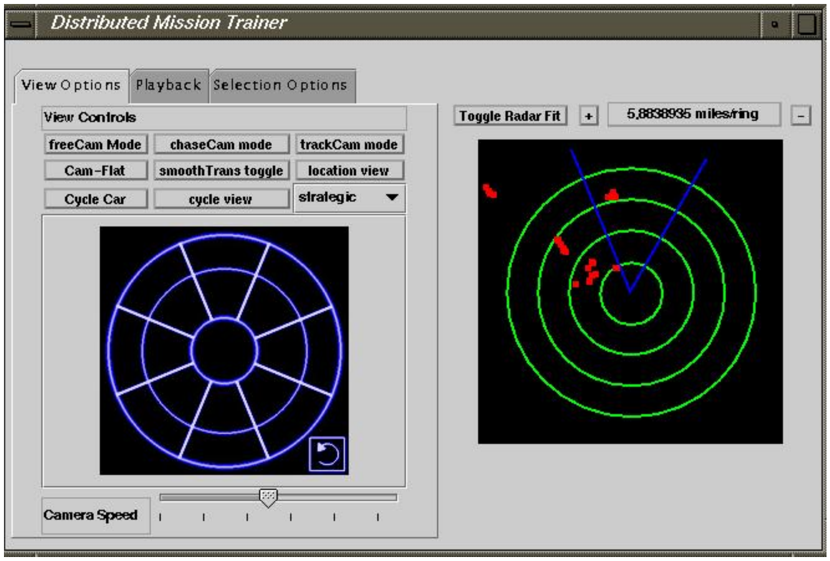

Figure 11: View management controls

Figure 11 shows the view management controls provided to the operator. These nine controls are located in a 3x3 block just above the navigation circle. These controls were added in a newer version than the interface shown in Figure 9. The first row of controls determines the mode of the camera in the Virtual Battlespace. If "Free Cam Mode" is selected, the user may navigate around the environment freely; this is the default camera mode. If "Chase Cam" mode is selected, the user moves with the selected unit; if no unit is selected pushing this button does nothing. Finally, if "Track Cam" mode is selected, the user looks at the selected unit from a set of predefined camera locations, switching from location to location as the vehicle moves. A real world analog of this camera behavior occurs when watching the leader of an auto race on TV. The camera crew switches from camera to camera around the track to keep the lead car on the TV screen. Once again, if no unit is selected, pushing this button does nothing.

The second row of buttons includes the "Cam-Flat", "Smooth Trans Toggle", and "Location View" buttons. The "Cam Flat" button sets all the camera angles to zero. This is a handy option if the user has moved the camera into a strange rotational state. The "Smooth Trans Toggle" changes the camera between smooth translation and teleportation. In smooth translation, if the camera is moved from one location to another one (this often occurs in Track Cam mode), it travels between the two spots smoothly using quaternions. If teleportation is chosen, the camera immediately goes to the new location. The "Location View" button toggles between predefined (in a config file) locations for the camera to go to. This is essentially a way to jump between defined waypoints in the Virtual Battlespace.

The third row of buttons includes the "Cycle Car" and "Cycle View" buttons and the mode pull down. The "Cycle Car" button changes the camera mode to "Chase Cam" if needed and then attaches the user to a random different unit on the screen. The "Cycle View" button moves the camera around the currently followed unit to allow the user to switch between views of it from behind, above and the side. The mode pull down allows the user to change the mode of the Virtual Battlespace. There are two choices: strategic and pilot. In strategic mode units are aggregated and increased in size and the entire engagement is visible. Furthermore, the user is placed high in the air and icons represent all of the units. In pilot mode, the units are deaggregated if needed, changed to life scale and represented by realistic models.

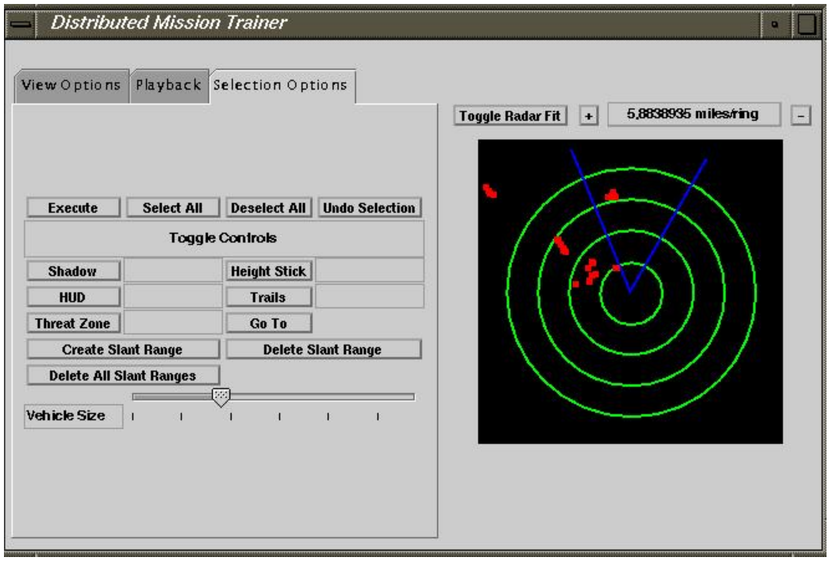

Figure 12: Selection and display management controls

Figure 12 shows the selection and display management controls. These controls allow the user to turn on or off all of the visual information that can be displayed for units in the Virtual Battlespace. Each of the display options is located in the “Toggle Controls” block of buttons on the left side of the interface. For each of the five toggles, the current state of the option is displayed in the box to the right. For example, if shadows were to be turned off, the box to the right of the “Shadow” button would say “OFF”. In this way the user can choose to engage or disengage shadows, height sticks, HUDs, trails and threat zones. Shadows appear beneath the units to show where over the ground they are. Height sticks are poles attached between the ground and the unit to show the altitude of the unit. The HUD is the heads up display shown by that unit and provides state information about the unit such as speed and heading. Trails are triangles left on the ground where units have been and are used to see the flow of the units across the battlefield. Threat zones are graphical representations of the effective range of threats such as SAM sites. When the “Execute” button is pressed, the options queued up in the toggles are applied to all selected units. The “Vehicle Size” slider allows the user to increase or decrease the scale of the units displayed in the Virtual Battlespace. The final graphical tool is the creation of slant ranges. A slant range is a line between two units. The “Create Slant Range” button creates slant ranges between all selected units. The “Delete Slant Range” deletes the last created slant range while the “Delete All Slant Ranges” button deletes them all.

The selection options are shown just above the “Toggle Controls”. These controls provide alternatives to using the radar shown to the right for common cases. The first of these is selecting all the units in the battle, which is accomplished by the “Select All” button. The “Deselect All” button deselects all of the units in the battle. “Undo Selection” removes the last selection made. The “Go To” button moves the viewpoint to the selected vehicle. Individual unit selection is accomplished by clicking the unit in the radar screen.

The conclusion from using it for two years is that the Java-based interface complements the typical immersive interface well. The display devices are portable and non-intrusive yet provide crisp display of detailed information. The Java-based interface can support much greater complexity and yet remain very intuitive to a user because it uses familiar paradigms displayed on a device familiar to the user.

Application Launcher

The Virtual Battlespace was designed to support multiple modes of use. In one case, the user will want to play back a recorded scenario and in another case they will want to interact with a live feed. Furthermore, the user will want to run the application in several different locations. Instead of having several different scripts to accomplish this flexibility, we designed an application launcher creatively dubbed the Launcher. Figure 13 shows the window that appears when the Launcher is run.

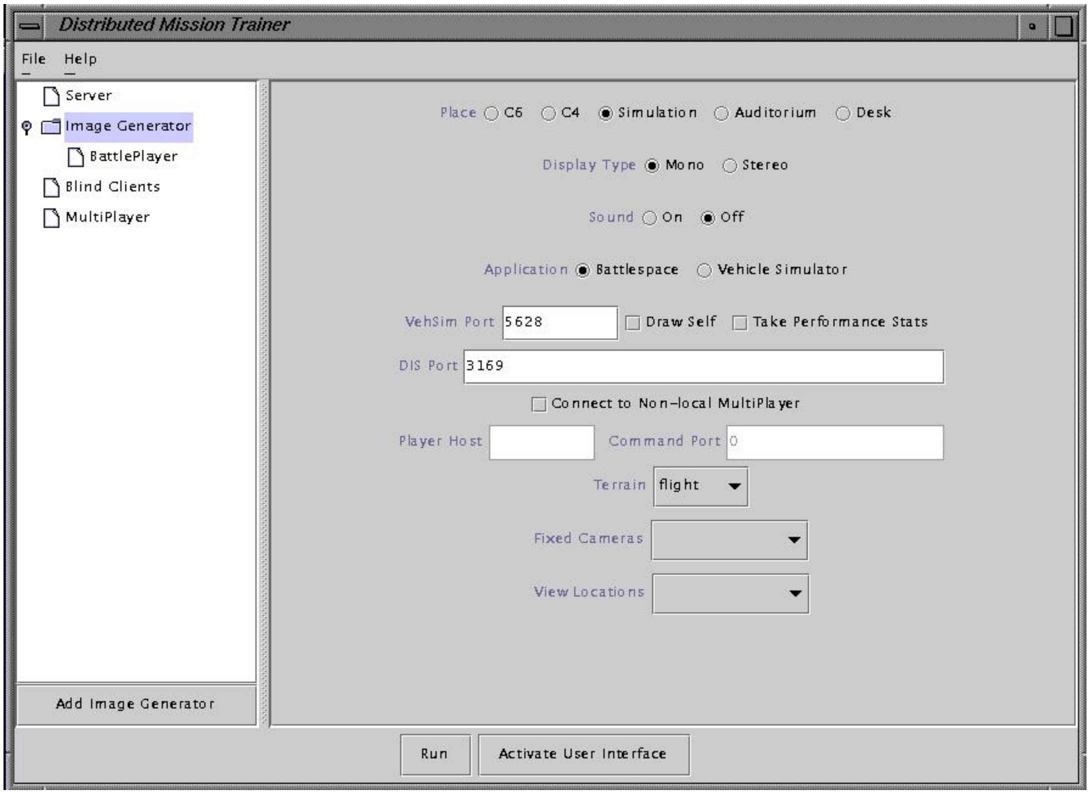

Figure 13: Launcher Connection screen

The panel on the left side of the Launcher shows what applications you are launching. In the case shown in Figure 13, the user is launching a server, image generator (with a battle player), blind clients, and a multiplayer. The image generator is the Virtual Battlespace and the battle player sends information to it about the units in the engagement. The multiplayer broadcasts the battle player's information. Blind clients run a flight dynamics engine designed here at Iowa State University and the server sends their state information to all clients (including the Virtual Battlespace). To add more applications, simply click one of the types you want to add and the button at the bottom of the left panel will change to add that kind of application.

The panel on the right side gives the options for the application selected in the left panel. Figure 13 shows the options for the Virtual Battlespace image generator. The first choice is location and these are several of the display devices here at Iowa State University discussed later. The second choice involves using stereo graphics or displaying in mono. The next option is used to run the application with or without sound. The next option chooses between running a Virtual Battlespace or a Vehicle Simulator. Most of the time, Virtual Battlespace would be chosen here unless this image generator is going to be for someone flying the locally designed flight simulator. In this case, Vehicle Simulator would be chosen and the user would control one specific unit in the engagement. The next option is the port number to the server, followed by

whether or not to draw the unit the operator controls and whether or not to take performance statistics of the application as it runs. The next choice is the port number to the battle player. Below that is a choice to use a multiplayer that does not reside on the local machine. If this box is selected the IP address of the machine and the port used must be entered in the boxes on the next line. The terrain to be used is selected in the pull down menu on the next line. The next option is the file that contains the locations of the Track Cameras for use in Track Cam mode. The final option in the right pane is the file that contains the locations of the waypoints for use with the "Location View" button (see Figure 11).

The two buttons on the bottom are the execution buttons. Once the user has configured the applications that they wish to run, the user must push the "Run" button. If the user also wishes to run the Tweek interface on the local computer, they must press the "Activate User Interface" button.

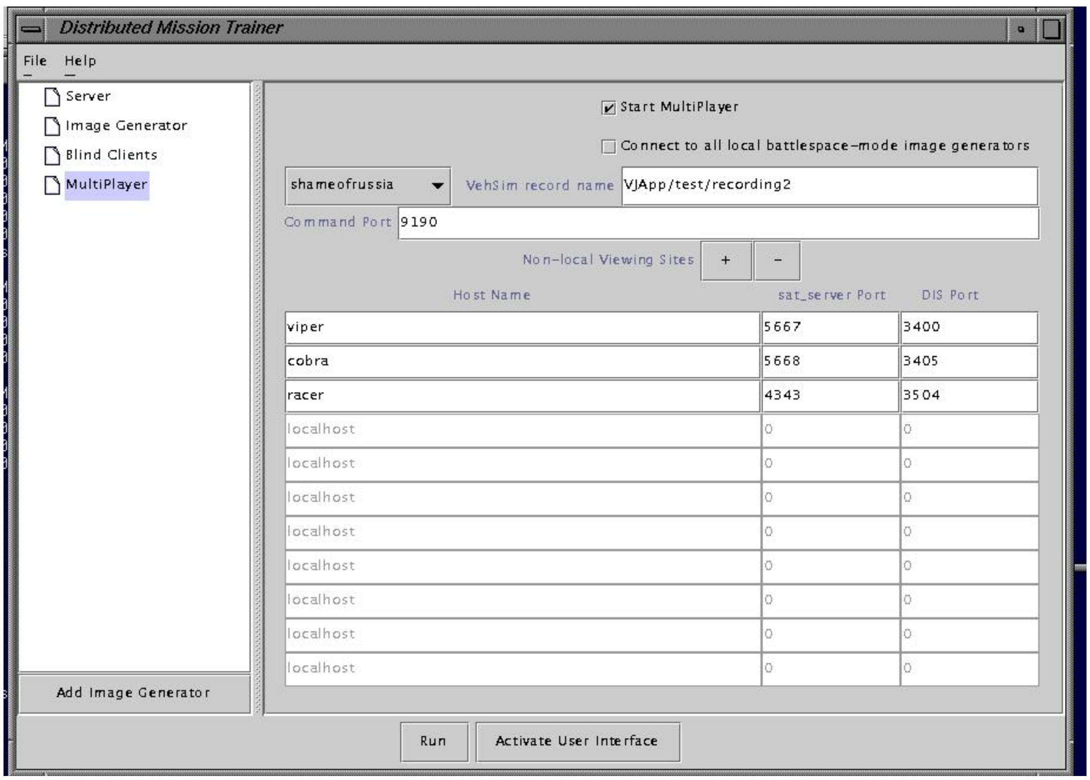

Figure 14: Launcher Multiplayer Management Screen

Figure 14 shows the option screen for the multiplayer. The first choice is whether or not the multiplayer should be run on the local machine and the second choice is whether or not this multiplayer should connect to all the Virtual Battlespace image generators it can find on the local machine. The pull down on the next line is the file with the battle scenario in it. The VehSim record name box is the file that contains a scenario involving units controlled by the flight simulator developed here at Iowa State University. The next line is the communication port for the multiplayer. The rest of the panel involves listing Virtual Battlespace image generators on other computers that need to be connected to this multiplayer. Pressing the "+" button adds another computer and "-" removes one. Each added computer shows up in the next available row of the table below the "+" and "-" buttons. Each computer is given an IP

address (or name), a port for their VehSim server, and a port for communication with the battle player. Screens such as this one are also included for blind clients and the server.

The Launcher makes running the many different configurations of the Virtual Battlespace application and its many related applications easy.   Without it, the user would have to maintain several different scripts for the different configuration options. This would result in an unmanageable number of scripts and a difficult to maintain execution system.

System Implementation

Virtual Battlespace is a VRJuggler application [11]. VRJuggler is a platform for the development of virtual reality applications that provides developers with the ability to use a single source code base to support a broad range of VR devices, from desktops and head-mounted displays to Powerwalls and Caves. VRJuggler abstracts I/O devices to allow the applications developer to focus the application and not the VR device configuration. VRJuggler is offered under an open source license.

Since it is built on VRJuggler, the Virtual Battlespace supports all of the immersive display devices found at the Virtual Reality Applications Center (VRAC) at Iowa State University (http://www.vrac.iastate.edu). In addition to desktop and head-mounted displays, VRAC has several large-scale immersive environments that have been used as test beds for the Virtual Battlespace.

VRAC's most immersive device is the C6 (Figure 15), a 10'×10'×10' room on which stereo images can be projected on all four walls, and the floor and ceiling [12]. The result is a totally immersive 360° field of view. The C6 is driven by a SGI InfiniteReality2 system and achieves a frame rate of approximately 40 Hz. Users inside of the C6 are tracked by a wireless Ascension Flock-of-Birds tracking system. The wireless tracking system leaves the user free to move about untethered. Virtual Battlespace in this type of environment would be used as a command station far removed from the battlefield. Perhaps planning the overall strategic direction of a conflict could be done from a device such as this. In addition, this type of installation would be useful for training exercises held far away from enemy lines.

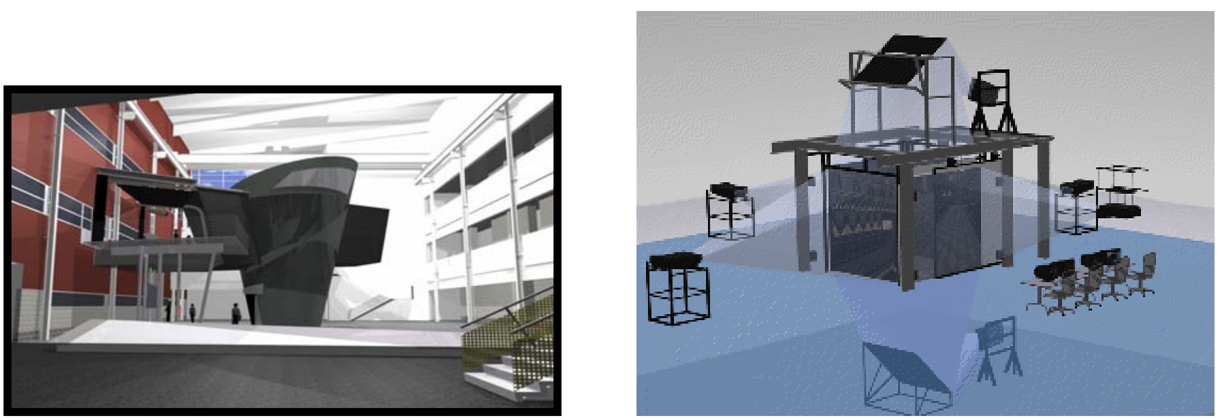

Figure 15: The C6 immersive display device

The next highest level of immersion available is the C4. The C4, shown below in Figure 16, has 3 walls and a floor. Additionally the walls can fold out to form a 36' power wall or fold in to form a closed environment. The flex system allows multiple different configurations to be available. Virtual Battlespace on this type of environment could be deployed in safe regions of an engagement area. This environment is still very immersive so the user could still take full advantage of the environment to plan engagements or monitor battle conditions.

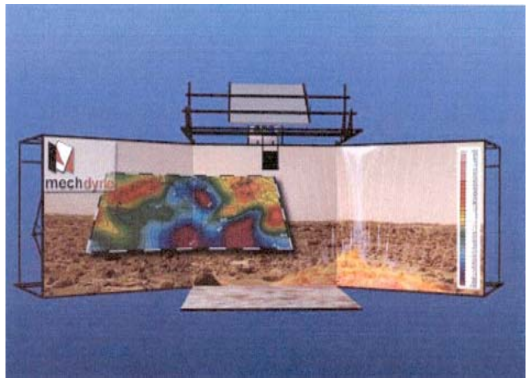

Figure 16: The C4.

Going down one level further, there is the power wall. The power wall provides 3D stereo graphics on one surface. It is a reasonable low cost solution to the display needs of the Virtual Battlespace. The power wall does not provide the same type of feel for the Virtual Battlespace as the more immersive environments. However, it does provide a user with a good 3D window into the battle environment or mission. This type of system could be deployed to training areas and to forward positions. See Figure 17 for an example power wall.

Figure 17: Barco Workbench

Lastly, the Virtual Battlespace could be run in our 300+ seat auditorium that is 3D stereo graphics enabled shown in Figure 18. This environment would allow a large gathering of people to view the same environment. Virtual Battlespace in this environment could be used for training, briefing, debriefing of a large number of people. It would allow everyone to have a common vision for the engagement that happened already or will be happening.

Figure 18: Lee-Liu/Alliant Energy Auditorium

In addition to the image generation resources required by the Virtual Battlespace are the networked computing resources that generate the various streams of incoming data. For example, VehSim streams representing individual ground vehicles and aircraft are generated by Windows-based vehicle dynamics engines, while the JSAF forces may be simulated on either Linux or Irix resources. The Virtual Battlespace supports a wide range of input devices including, for example:

- a Microsoft Sidewinder Steering wheel and pedals for ground vehicles,
- a Microsoft Joystick for air vehicles,
- a variety of physical bucks for ground or air vehicles, and
- several wireless-enabled personal interface devices (PDAs and Tablet PCs).

Features

Consider a scenario involving an engagement between Red team and Blue team. Blue team is tasked with destroying Red team's headquarters located in Nellis Air Force range in Nevada. Two SAM sites and five squadrons of fighter aircraft defend the Red team headquarters. Blue force consists of seven groups of aircraft. When the engagement is viewed strategically, these groups of aircraft are shown as aggregate entities and are scaled greatly to be visible from a long distance. The aircraft aggregates appear as symbolic entities but are placed in the space at the correct position height.

When the application starts, the user is presented with a view that encompasses the entire engagement. In addition to the terrain and the units engaged, the user is also presented with an information "billboard"—so called because it appears across the top of the display no matter where the user navigates (see Figure 19).

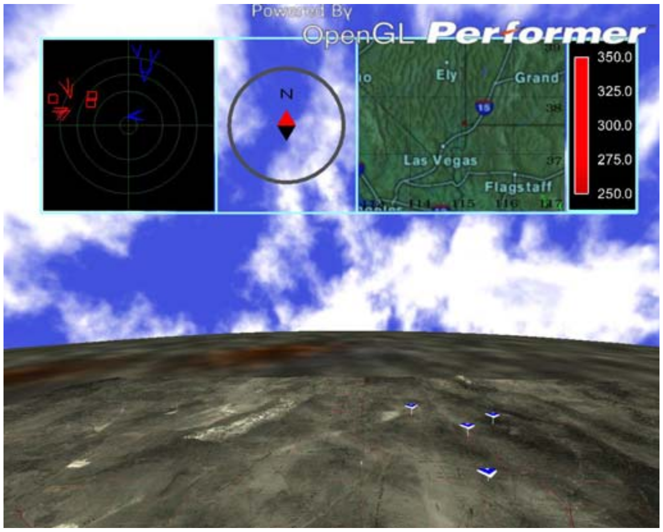

Figure 19: Billboard battle information display

The billboard allows for the presentation of multiple simultaneous information channels. These may include symbolic views of the battlespace, such as synthetic “radar” screens, maps indicating additional features of the battlespace not contained in the main terrain display, orientation aides, and graphical keys.

This other information actually can be quite significant. Video feeds relating to the battle can be integrated into the billboard informing the user directly of events that are happening or that have happened and require attention. In Figure 19 we show four different types of information streams that we have chosen to display on the billboard. The entity radar, the leftmost piece, allows the user to maintain constant awareness of the position of all of the entities. The radar always has north as up and so also give the user a frame of reference to understand the angles of the battle that they wish to see. The second to the left is the compass. The compass helps the user maintain a sense of what direction is north. If the needle is pointing straight up then the user is facing north. It is also useful in orientating the user. The second from the right is political position map. This map indicates where the user is in the world from a two dimensional standpoint. This presents a different avenue for the user to understand their location and reinforces the users situational awareness. The furthest right piece of the billboard is the speed indicator. Based on the color of the history trails of each entity the graphic indicates how fast the group of entities is traveling. This allows the user to grasp data about a group of entities without having to go to the entity specifically.

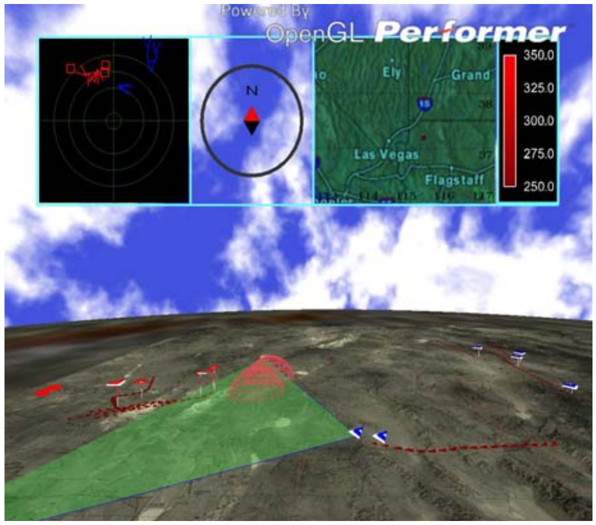

Figure 20: Graphical Information in the Battlespace.

The individual entities use a variety of graphical methods to display information about their status. For example, in addition to position, orientation, and velocity, entities in the space can also leave a colored trail indicating where they have been or where they may be targeted to go. Through the decoupled Java-based interface, one or more users control the configuration of these additional display mechanisms. Using this interface, the user is able to navigate through the battle and focus on areas of interest. The interface can also be used to select entities by position, call sign, or type, and reconfigured to display additional attributes. For example, as shown in Figure 20, the Blue team lead sensors sweep reveals which Red team units are within the range of the Blue team's "vision". Also visible in Figure 20 are the threat domes and history trails of different entities. SAM sites can have their threat envelopes turned on which display as red wire mesh domes. Also entity history trails give the user an idea of where entities have been and possibly a prediction on where they will be in the future.

Virtual Battlespace incorporates a variety of points of view to allow users to gain useful perspectives on simulated engagements. Figure 21 depicts the battle from a long-range (or strategic) point of view. Units are displayed symbolically at a size consistent with the unit's importance, rather than its physical distance.

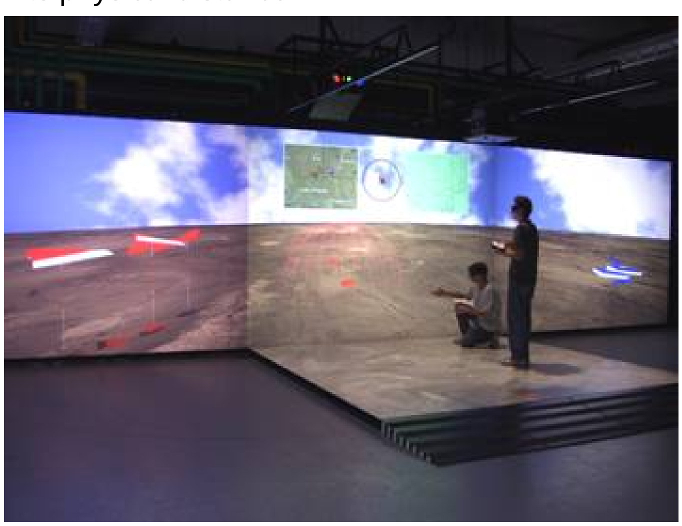

Figure 21: C4 Strategic Battlespace

Figure 22 shows an alternative view that combines a realistic first-person entity perspective with symbolic information. This allows a user to adopt a tactical perspective combining the participant's first-person view with battle-level sensor information or other abstractions. This means that a user could view the battlefield from the point of a view of a particular squadron and still see the threat domes and sensor sweeps of other entities.

Figure 22: First-person view.

RESULTS AND DISCUSSION

Systems Training Exercise

As a proof of concept, the Virtual Battlespace was incorporated into a training exercise of the 133rd ACS in August of 2001. The details of this test can be found in [13]. Operators in the OM interacted with a simulated battlespace generated from a JSAF simulation. JSAF was chosen as the force generator in this case, although any DIS compatible computer force generator could have been chosen. Trainers at the 133rd constructed a flexible scenario incorporating a set of autonomous entities that followed predefined missions. These were augmented with additional units, controlled by role players, to increase the level of interactivity of the simulation. The simulation entities did not pursue their missions in isolation, but were capable of coordination based on expert behaviors programmed into the system. The output of this combined simulation was converted to a DIS stream and used to stimulate the OM modules. This same DIS stream was captured by a Virtual Battlespace recorder for later replay within the Virtual Battlespace.

A series of operators in the OM interacted with the MCE two-dimensional radar displays to control a collection of more than 50 entities. They identified tracks, communicated with battle managers and simulated pilots while conducting a control exercise for several hours. The recorded DIS stream was delivered the following day to VRAC where it was replayed in a variety of contexts. A large group of guard observers participated in a post-exercise debriefing in VRAC’s stereo visualization auditorium, reviewing the day's exercise from a variety of perspectives. In addition, small groups interacted with the captured simulation in the totally immersive six-sided C6 projection cube.

The exercise showed that use of both MCE stimulation and VR based review provided a more complete training and debrief capability for C2 operators. One of the strongest suggestions arising from this exercise was a focus on deployability of the capability, which was addressed by the development of alternative versions suitable for display on low-cost commodity hardware systems.

Deployability

The initial visualization system developed for this research required high-end display and image generation hardware. Developed under the Silicon Graphics (SGI) IRIX OS, using an SGI Reality Engine supercomputer as an image generator and multiple high-resolution digital projectors, the system taught us much about how immersion can be used for battlefield visualization. However, as implemented, the system required fixed resources and was not deployable.

At a September 2002 meeting at the 133rd Dr. Brooks and Col. Breitbach reaffirmed their interest in a deployable immersive capability to support exercises in the field. They felt that providing a Joint Forces Commander with the ability to "see" the battlespace would facilitate the decision making process, but it would require a technology that could be deployed quickly and easily, with a small footprint.

With their new GTACS training system, Col. Breitbach and the 133rd have shown that Commercial off-the shelf (COTS) software and hardware can be combined to create very capable systems at comparatively low cost. In that spirit, we investigated an alternative software system for battlespace visualization suitable for deployment on a display system based on

commodity hardware. Such a system could be used to enhance the training, and ultimately the operational, capabilities of command and control units like the 133rd.

An added benefit of this deployable development was the ability to more easily display battle control technology in public forums and conferences. The rich visualization deployed on a portable display, with a low-cost but powerful image generator, provides a solid overall context for other facets of distributed mission training, such as flight simulators, or individual mission control stations.

One of the PIs, Dr. Cruz, has been working in the area of reconfigurable displays for several years. In 2002, Dr. Cruz developed a design for a low-cost system that was both portable and reconfigurable. The work resulted in the deployment of the system shown in Figure 23 at the VRAC. Each projection surface is a self-standing and self-contained module that integrates the screen, projectors and computers. Figure 26 shows one of these modules. The frame was designed to vertically match among the modules and to stand on coasters, so multiple configurations could be set with little effort. Some of these configurations are shown in Figure 24 and Figure 25. Back to back configuration used at SIGGRAPH 2003 show several configurations used at different professional conferences.

At SIGGRAPH 2003, the screen layout was changed twice a day every day of the show -- from a 2-wall theater configuration to a back to back configuration. The same display modules were used to show widescreen applications and also have two demonstration stations to showcase collaborative immersive spaces. The module reconfiguration takes about 15 minutes, with much of that time going to rebooting the computers once the modules are moved.

This portable system has proven very useful for our research team, allowing us to bring our research results to a variety of events. However, this system suffers from the same resolution limitations as other immersive systems. We hope to use the lessons learned and experience gained in developing this system to inform and improve the design and construction of the system proposed here.

Figure 23: "Baby" cave

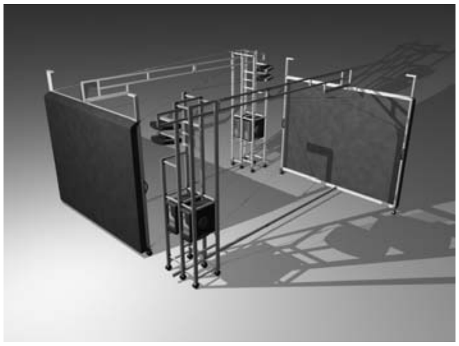

Figure 24: Theater configuration used at SC 2002

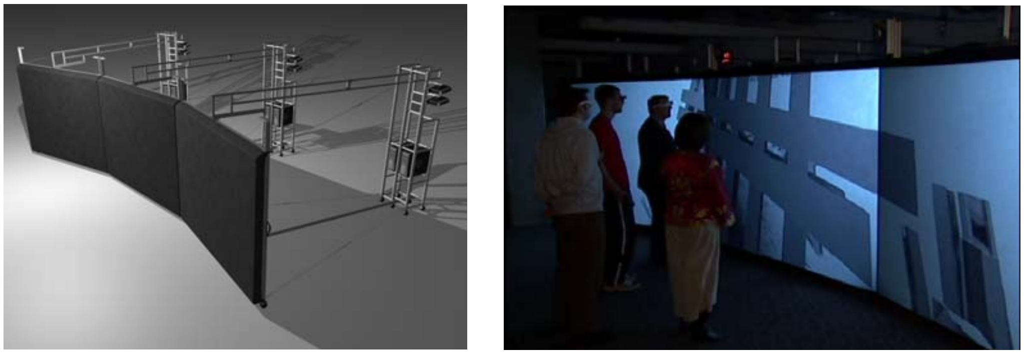

Figure 25: Theater configuration used at SC 2002

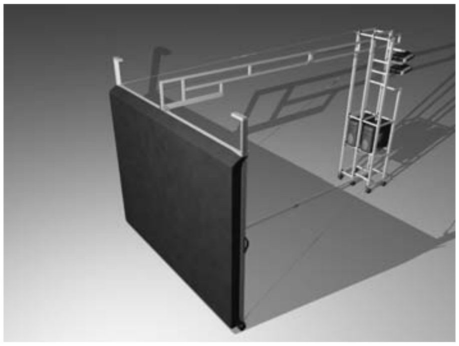

Figure 26: Self-Contained Module

With the "Baby Cave", we were able to demonstrate that immersive visualization could be deployed without the use of expensive fixed assets. However display hardware is only part of the deployability picture. To accomplish deployability, image generation and system simulation for the Virtual Battlespace had to be moved from the expensive shared memory system it was developed on to a less-expensive, less special purpose computing platform.

While commodity personal computers have increased in power dramatically over the past several years, they are still not sufficiently powerful to run complex, multi-channel immersive projection systems. To accomplish this using PCs, one needs a cluster of PCs, which can coordinate to run an application synchronously to feed the multiple channels characteristic of immersive display. PC clusters are a lower cost alternative to traditional shared-memory multiprocessor supercomputers, but the synchronous frame generation demands of virtual reality applications complicate their application to computer graphics.

Fortunately, VRAC is home to the development team of VR Juggler, an application development platform for the creation of virtual reality applications. During 2002 and 2003, one of the project PI's, Dr. Cruz, led her team in the creation of an extension to the VR Juggler platform called Cluster Juggler. Cluster Juggler

> *"… is designed to make PC clusters a feasible alternative to expensive shared memory systems. It cannot simply utilize the design of traditional clusters because virtual reality requires special functionality not present in conventional cluster applications. It also should not put the burden on the VR application developer to perform communication between cluster nodes. To prevent burdening the developer, the design retains some of the features that shared memory systems provide for virtual reality by hiding the complexities of a cluster. Our goal is to design a distributed shared memory system for VR application I/O (input/output) data. As a result of distributed I/O, application development and execution can transparently move between shared memory VR systems and PC cluster VR systems. This means the same VR applications will run on both systems, with no or very minimal changes." [14,15]*

ClusterJuggler provides abstractions to allow application developers to remain unaware of the complexity of input and rendering management that running on a coordinated cluster present. In virtual reality applications the frame loop is a primary driver, and maintaining coordination of that frame loop on a machine by machine basis goes a long way to ensuring that applications running on the individual cluster nodes remain in sync. Ensuring coordinated delivery of input events is also a critical dimension of synchronization. By providing abstractions to encapsulate these critical functions, ClusterJuggler is able to guarantee synchronization when its primitives are used correctly, relieving the burden on the application programmer to manage inter-node coordination. Figure 27 below shows the architecture of a Juggler application.

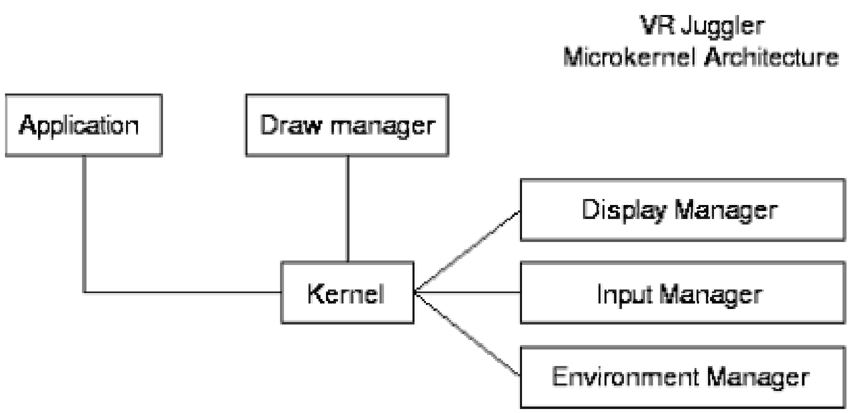

Figure 27. VR Juggler Microkernel Architecture

ClusterJuggler extends the input manager to coordinate communication between cluster nodes as input is received. The input manager also provides a locus for input pre-processing to allow transformation of input prior to per-node processing. This can remove redundant work from the distributed system and depending on the nature of the processing can be used to compress the input exchanged between nodes over the network.

In addition to input synchronization, ClusterJuggler provides an abstraction for coordinating the multiple displays characteristic of immersive displays. Though individual PCs do not typically

have sufficient processing power or graphical processing units to support the multiple channels inherent in immersive displays, clusters of PCs can provide this capability if the frames that each PC generates can be synchronized. This amounts to ensuring that no frame is displayed by a node until the slowest node can display that frame. ClusterJuggler is capable of supporting hardware or software blocking. Applications are synchronized at startup, input arrival is synchronized, and frames are coordinated to ensure that each copy of the application generates a scene from an identical state.

Figure 28 shows the Virtual Battlespace being run at I/ITSEC 2002 by Dell PCs running Linux.

Figure 28: Linux prototype of Battlespace on portable stereo display wall

Applications of the Virtual Battlespace

Immersive visualization that provides users with an integrated view of both the visible and the invisible attributes of an ongoing engagement has many possible applications, in training, planning and even some day operations. The ability to fuse various streams of information from the battlefield into a common, integrated display promise to make visualizations like the Virtual Battlespace a useful tool for training and debriefing, real time command and control, the operation and control of unmanned vehicles, and strategy planning. The following sections discuss how the Virtual Battlespace can be a key component of each of these applications.

Training and Debriefing

Immersive Battlespaces can be the basis for valuable training for both weapons directors and battle managers. The Virtual Battlespace provides an overview of an entire engagement in an immersive environment making it ideal for distributed mission training. Participants in a Battlespace training environment are able to see each event and can make decisions based on all of the pertinent battle information. In addition, the immersive nature of the Virtual Battlespace provides an advantage to battle commanders since they are able to look around in three dimensions to gain information as if they were standing in the middle of the battlefield. In

comparison, the commander would have to either use multiple screens to see in all directions or scroll between screens displaying only a part of the battle. This enhances the usefulness of the Virtual Battlespace as a training tool beyond that gained by desktop simulators. Furthermore, since the Virtual Battlespace is a simulation, different scenarios can easily be tested without the high cost of actually deploying units. Indeed by networking Virtual Battlespace simulations together, one commander can be pitted against another to create a war game. Weapons director training has an additional advantage of note: the ability to show threat ranges and threat volumes of any weapon in the simulation. This information is especially valuable to weapons directors as it allows them to know at any time which weapons could hit specific targets. The Virtual Battlespace can also be a useful tool for pilots and squadron leaders by connecting high fidelity flight simulators. This could result in several different squadrons interacting in a shared space, providing an ideal environment to train group tactics both within a squadron and between squads.

An integral part of training is briefing and debriefing, which the Virtual Battlespace is also able to accommodate. Prior to a training exercise, a pre-run of the simulation can be displayed to inform participants of the characteristics of the upcoming scenario. As the engagement is fought, the Battlepsace can be used to track progress and all the decisions made by the users as well as the positions of all the battle participants can be recorded. After the exercise, the entire simulated engagement can be replayed interactively, allowing further analysis of decisions with trainees after the battle as many times as needed, from any perspective, and at any speed desired.

Real Time Command and Control

The ability to quickly ascertain a detailed visualization of the battlefield situation suggests that visualization such as the Virtual Battlespace will be an important component of the command and control station of the future. Currently, commanders rely on separate displays each representing a stream of information such as radar, communications, and pre-battle plans. The integration of all this data is accomplished by utilizing the commander’s mental imagery. Therefore, commanders must always spend effort keeping an accurate and up to date image of the engagement in their heads while trying to make decisions about how to act upon this information. This demanding mental workload results in longer delays between decisions. By using depictions like the Virtual Battlespace, disparate information streams can be fused and displayed in an immersive format for the commander. This would allow commanders to focus on making quick and effective decisions. Additionally, if a commander were trained using the Virtual Battlespace, its use in a real engagement would seem very natural and desired over other methods of control.

Operation of Unmanned Vehicles

A current desire of the armed forces is to reduce the number of people in harm’s way by using unmanned vehicles. While for the most part unmanned vehicles are not ready for real combat, there is one notable exception, the Unmanned Combat Aerial Vehicle (UCAV). UCAVs provide valuable data through high-resolution photography while putting no human operator at risk. Unfortunately, UCAVs require a team of people to fly. There are multiple video camera feeds and radar information sources that must be integrated in the minds of the operators.

Instead of controlling the UCAV with all of these different displays, a high fidelity flight simulator updated by these feeds would provide a more natural method of control. It would give the operator the illusion of piloting the vehicle. This would reduce the mental workload of the operator so that one person could fly the plane. If this simulator were joined with the Virtual Battlespace, extra information such as threat zones of enemy weaponry could be made visible. In addition, the pilot of the UCAV could briefly separate from the plane to gain an overview of the entire battle as it is happening.

Another troublesome characteristic of how UCAVs are currently piloted is that the video feeds and user commands are lagged. If this delay is significant, the vehicle will not be able to respond to pilot inputs quickly enough. Additionally, the pilot may be flying with an image of where the vehicle used to be rather than where it is now. The Virtual Battlespace could compensate for this lag by simulating all of the vehicles detected by radar forward in time by the known lag so that the real time position of all battle participants can be approximated, including the UCAV itself. In fact, if the lag can be estimated, the Virtual Battlespace can use a sophisticated dead reckoning method using the history of all the commands the pilot has entered since the position was sent by the UCAV to estimate where it is at the current time. By estimating both the position of the UCAV and the other participants in the battle, the Virtual Battlespace could provide a more up to date view of the environment surrounding the UCAV.

Strategy Planning

Planning for an engagement begins well before the first action is taken and is a process that requires significant resources and time. Once a plan is adopted, even if it can be changed, it is costly in money, resources and time to change. If commanders were to use the Virtual Battlespace as a planning tool, they would be able to play out a scenario as soon as it was conceived, immediately gaining useful information about it. The Virtual Battlespace also saves the scenarios so they could be recalled instantly and modified to fit a new situation. In addition, strategies could be developed in real time by placing and commanding units in a simulated battle while instructing the Virtual Battlespace to record all of the commands. In this way, battle strategies could be generated and not just reviewed for effectiveness.

Future Direction

Based on our experience developing the Virtual Battlespace, we believe that one of the most promising future directions for the use of immersive military visualization is in its application to the operational command and control of UAVs and UCAVs.

The complexity and capability of UAVs (such as the Predator shown in Figure 29) is expanding rapidly and the range of missions they are designed to support is growing. By 2012, the DOD UAV roadmap projects that F-16-size UAVs will perform a complete range of combat and combat support missions, including Suppression of Enemy Air Defenses (SEAD), Electronic Attack (EA), and even deep strike interdiction [16]. UAVs specialize in missions commonly categorized as “the dull, the dirty, and the dangerous”. As such, they promise to be effective force multipliers that preserve the lives of military personnel.

Figure 29: RQ-1 Predator

Figure 30: RQ-1 Predator ground control station

For the UAV's potential to be reached, significant technical issues must be overcome. Several of these challenges are human interface issues, related to the systems used to command and control UAVs in a mission such as the control station shown in Figure 30. Chief among these is to develop new operational control systems that expand the situational awareness of the operator beyond the level provided by today's "soda straw" optical systems. [17]

According to the DOD Roadmap, the ground control station – the human operator's portal to the UAV – must evolve as UAV's grow in autonomy. The ground control station will facilitate the transformation of the human from pilot, to operator, to supervisor as the level of interaction with the UAV(s) moves to ever higher levels. As the human interfaces with the UAVs at higher levels, the human must trust the UAV to do more. To develop and maintain that trust, the human must be able to understand the intent of the UAV in the overall mission context and monitor its performance. Further, these next generation interfaces must allow for an operator to assume and relinquish direct control over a managed vehicle multiple times during the course of a mission. To do this effectively, operators must be able to quickly develop a precise understanding of a vehicle's operational context.

The challenge of designing an effective UAV control interface is made more difficult by the desire to control groups of UAVs. These groups "must be controllable by non-specialist operators whose primary job is something other than controlling the UAV. This demands … a highly simple and intuitive control interface … and the capability for autonomous vehicle operation of one or more vehicles being controlled by a single operator" [16]. The goal for these interfaces is to increase the human operator's span of control while decreasing the manpower needed to operate any one vehicle. Coordinated advances in the vehicles and the command and control interfaces used to supervise them will be required to accomplish this goal.

Multi-vehicle operator control systems will need to provide far more comprehensive information than present systems on the state of the overall mission during normal operation of a semi-autonomous swarm of vehicles. Furthermore, these systems must be capable of directing the operator's attention to emergency conditions and provide him or her with the context needed to effectively assume direct control of an individual aircraft if necessary. These advanced interfaces will have to fuse all of the information needed by the pilot into the view used for vehicle control. They should also take advantage of as many senses as possible, including force feedback and aural cues to provide more avenues for the presentation of information.

To meet this challenge, we propose a new design for an immersive ground control station that will provide operators with the comprehensive context and control flexibility necessary to effectively monitor and control one or more semi-autonomous unmanned remote vehicles. This new design utilizes a virtual reality based visualization of the operational space, a synthetic graphical representation fusing multiple information streams into a comprehensive immersive environment designed to significantly enhance an operator's situational awareness. The environment will simultaneously inform the operator about the position and condition of the vehicles under his or her control while providing an organizing context for all available information relevant to the engagement.

We believe that the technology exists to create and study working prototypes for these advanced interfaces based on immersive synthetic visualization. Our work in this area is motivated by the combination of two related projects: the battlespace visualization research that is the primary subject of this report, and an NSF funded project in virtual reality aided teleoperation. While developing the battlespace, researchers at VRAC also began work on a new teleoperation control system that combines vehicle dynamics simulation, position and orientation tracking, and a virtual reality representation of the operational environment to create a vehicle control station that provides superior situational awareness and vehicle control in the presence of signal lag [18,19]. The primary components of this new VR aided teleoperation system are shown below in Figure 31.

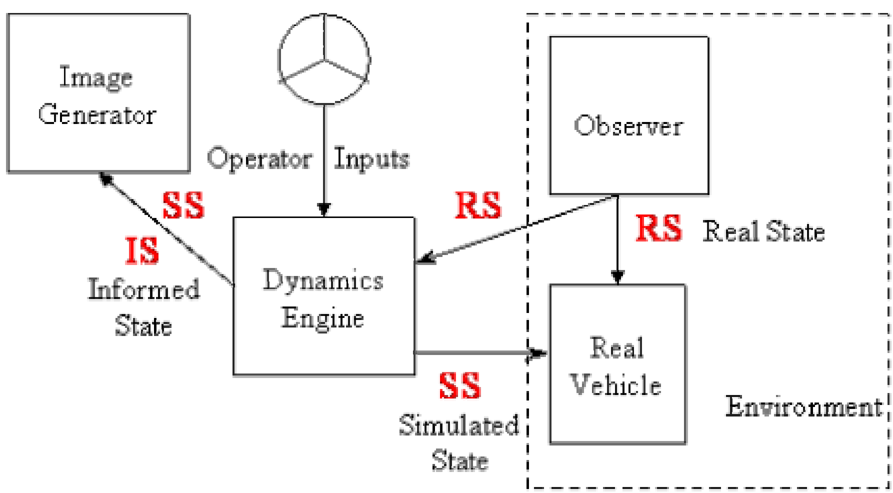

Figure 31: VR Aided Teleoperation Layout

The operator controls the vehicle from a virtual environment displayed by the Image Generator using an immersive control interface. The operator's commands are sent to a dynamics simulation; the simulation uses these inputs to predict the dynamic state of the virtual vehicle. The dynamic state represents critical state variables such as the vehicle's position, velocity, acceleration and heading. The simulated state created by the dynamics engine is used to both position the virtual vehicle and to provide a desired path for the teleoperated vehicle.

As these simulated states are received by the teleoperated vehicle, they are synchronized to account for lag and jitter generated by the communications delay. The vehicle uses these synchronized simulated states as a series of goal states. A simulation run locally on the vehicle determines the inputs required to get the vehicle to approach the simulated state from its current state. Of course, to calculate these approach paths, the current state of the vehicle must be determined. A tracking or observer system provides this needed state information. The observer system is responsible for asynchronously reporting vehicle states to both the operator and the vehicle itself. The operator uses the reported vehicle position, corrected for lag and

subsequent vehicle control, to visualize likely future positions of the vehicle. These predicted positions could be depicted graphically to allow operators to adjust their control to obtain higher fidelity with the remote vehicle, closing the loop between the human and the computer controlling the remote vehicle.

VR aided teleoperation has the potential to significantly expand the capability of current generation UAV control systems. The system's synthetic operational picture allows far more sources of information to be integrated into a single picture than displays based around vehicle video feeds. In addition to the views available from onboard cameras, the radar-derived positions of enemy and friendly units, as well as unidentified tracks can be depicted in the same virtual world. Terrain in this view can be generated as a composite that fuses satellite imagery, political boundary maps, DTED data and other sources. The obstacles to creating useful terrain models are not unreasonably high. Since only general landmark features are necessary to fly a UAV, the terrain model is not required to match every contour of the real terrain to be effective. Furthermore, the immediate environment for most UAV's is the sky, so the UAV operator's primary concern is avoiding other aircraft. The virtual approach allows all of the relevant data streams to be gathered and displayed in a single operating picture, providing the UAV operator with a more comprehensive situational awareness of his operating environment than present generation systems.

The ground control station for multiple UAVs can be created by extending the VR based control station for a single UAV. A swarm manager would be placed in a large display system, such as a CAVE or widescreen display. The manager would then be presented an overall battle view similar to the highest scale provided in the Virtual Battlespace. With this overall view, all of the battle participants could be monitored, including the UAV swarm. Additionally, the manager would be capable of selecting any one of the UAVs in the swarm to adopt the "first person" perspective of that UAV. The UAV swarm would be controlled using the VR aided teleoperation system. With this system, the manager could issue commands directly to any UAV and those commands would drive the simulation, which in turn would modify the actual UAVs behavior. Furthermore, the manager could issue commands to the whole swarm or just a group of them. Any number of views and command structures could be incorporated in the virtual world. Finally, the semi-autonomous UAVs could send alerts to attract the operator's attention. An example would be when their control algorithms are overwhelmed or when they identify a target of interest. In this way, the virtual world can assist UAV operators by directing their attention to the most interesting or critical events occurring in the engagement.

REFERENCE

[1] W. Robertson, *The Staff Ride*, Prepared for the U.S. Army Center of Military History, Washington, D.C., CMH Pub 70-21, Supt. of Docs. no.: D114.2:R43, 1987.

[2] G. George, R. Brooks, H. Bell, R. Breitbach, R. Steffes, & C. Bruhl, (1999). "Synthetic forces in the Air Force command and control distributed mission training environment, current and future", *Proceedings of the Eighth Conference on Computer Generated Forces and Behavioral Representation,* Orlando, FL, 8, 319-331.

[3] D. Alberts, *Information Age Transformation: Getting to a 21st Century Military*, Washington, DC: CCRP Publication Series, June 2002.

[4] J. Posdamer, J. Dantone, N. Gershon, T. Hamburger and W. Page, "Battlespace Visualization: A Grand Challenge", *Proceedings of the IEEE Symposium on Information Visualization 2001 (INFOVIS'01)*, San Diego, CA, October 2001.

[5] J. Durbin, J. Swan II, B. Colbert, J. Crowe, R. King, T. King, C. Scannell, Z. Wartell and T. Welsh. "Battlefield Visualization on the Responsive Workbench", *Proceedings IEEE Visualization '98, IEEE Computer Society Press*, pp. 463–466, 1998.

[6] D. Hix, J. Swan II, J. Gabbard, M. McGee, J. Durbin and T. King, "User-Centered Design and Evaluation of a Real-Time Battlefield Visualization Virtual Environment", *Proceedings IEEE Virtual Reality '99, IEEE Computer Society Press*, pp. 96–103, 1999.

[7] B. Brehmer and C. Sundin, "Command and Control in Network-Centric Warfare", *Proceedings of Conference on Network-Centric Warfare*, London, June 23-24, 1999.

[8] G. Jense, N. Kuijpers and A. Dumay, "DIS and HLA: Connecting People, Simulations and Simulators in the Military, Space and Civil Domains", Simulations and Simulators in the Military, Space and Civil Domains, 48th International Astronautical Congress, Turin, Italy, October 6-10, 1997.

[9] O. Balling, M. Knight, B. Walter and A. Sannier, "Collaborative Driving Simulation", *SAE Paper No. 2002-01-1222*, February 2002.

[10] M. Mine, F. Brooks Jr., and C. Sequin, "Moving Objects in Space: Exploiting Proprioception in Virtual-Environment Interaction", *Proceedings of SIGGRAPH 97*, Los Angeles, CA

[11] Homepage of VRJuggler.org, http://www.vrjuggler.org.

[12] Fakespace Systems, http://www.fakespacesystems.com

[13] R. Breitbach, R. Brooks, G. George, R. Steffes, and S. Pegg, "Innovative C2 Training Solutions For Air Force Modular Control Systems", *International Training and Education Conference*, Lille, France, 2002

[14] E. Olson, Cluster Juggler—PC Cluster Virtual Reality, Masters Thesis, Iowa State University, 2002.

[15] Cluster Juggler information, http://www.vrjuggler.org/gadgeteer/0.8/clusterjuggler.guide/clusterjuggler.guide.html

[16] US Department of Defense, “Unmanned Aerial Vehicles Roadmap 2002-2027,” December 2002, http://www.acq.osd.mil/usd/uav_roadmap.pdf

[17] Charles L. Barry and Elihu Zimet, “UCAVs Technological, Policy, and Operational Challenges,” *Defense Horizons*, October 2001, Number 3, Center for Technology and National Security Policy, National Defense University.

[18] B. Walter, Virtual Reality Aided Teleoperation, Thesis, Iowa State University, Ames, IA, August 2003.

[19] J. Knutzon, Tracking and Control Design for a Virtual Reality Teleoperation System, Thesis, Iowa State University, Ames, IA, August 2003.